REMARKS

The Office Action dated June 21, 2004 stated that the claims should be restricted to either claims 1-11 and 22 or claims 12-21. The Examiner noted that in a telephonic conversation with Mr. Tim Sullivan, claims 1-11 and 22 were elected without traverse. In accordance with this conversation, applicants' attorney has reflected this election as set forth above by formally withdrawing claims 12-21.

I. Rejection of Claims 1-11 and 22 under 35 U.S.C. 112, second paragraph.

Claims 1-11 and 22 were rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicants regard as the invention. Specifically, the Office Action states that claim 1 is indefinite because it is unclear how the obtained predicate expression is used. The Office Action continues by stating that claim 1 is indefinite because it is not clear what is transformed. The Office Action further states that claim 1 is indefinite because it is not clear how the modified conditional branch is different from the original conditional branch. A similar rejection was made with regard to claim 22. Even though applicants believe that the claims are definite as written, claims 1 and 22 have been amended as set forth above to further clarify the invention and expedite allowance. Applicants assert that the aforementioned changes overcome the rejection under 35 U.S.C 112, second paragraph.

II. Rejection of Claims 1, 6 and 22 under 35 U.S.C. 102(e).

Claims 1, 6 and 22 were rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,446,258 issued to *McKinsey et al.* (hereinafter "*McKinsey*"). Claim 1 has been amended to recite the step of "employing runtime data that has been collected on a compiler to determine a binary-level conditional branch to be reversed." Independent claim 22 recites similar language as claim 1.

McKinsey teaches a method of compiling instructions of a program. *McKinsey* further teaches regenerating a predicate expression by an IF-conversion. (Col. 11, lines 4-5). The predicate expression can be regenerated via the IF-conversion to invert a conditional branch.

(Col. 11, lines 4-5). Contrary to the contentions set forth in the Office Action, *McKinsey* does not teach, show or otherwise suggest a computer-implemented method (medium) that operates at a binary-level to reverse a conditional branch within a binary program on a computer architecture that supports a predicated execution. Also, *McKinsey* does not teach employing runtime data that has been collected on a compiler to determine a binary-level conditional branch to be reversed. Accordingly, independent claims 1 and 22 cannot possibly be anticipated by *McKinsey*.

Regarding claim 6 of the present invention, claim 6 specifically recites, "determining the binary-level transformation comprises computing an inverse predicate expression that describes the opposite condition." Applicants assert that this limitation is not taught or otherwise suggested by the cited art. Furthermore, claim 6 depends from independent claim 1, which is clearly allowable as set forth above. Accordingly, applicants assert claim 6 is also allowable for at least those same reasons.

With regard to claims 2-5 and 7-11, the Office Action did not address those claims under 35 U.S.C. 102 or 103. The Office Action only rejected those claims under 35 U.S.C. 112, second paragraph. Insofar as the 35 U.S.C. 112, second paragraph rejection has been traversed, applicants assert that claims 2-5 and 7-11 are also in condition for allowance.

In view of the above amendments and remarks, Applicants respectfully request a Notice of Allowance. If the Examiner believes a telephone conference would advance the prosecution of this application, the Examiner is invited to telephone the undersigned at the below-listed telephone number.

Respectfully submitted,

MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, Minnesota 55402-0903
(612) 332-5300

Date: September 2, 2004

Ryan T. Grace
Reg. No. 52,956